INFORMATION AND CONTROL **67**, 158–162 (1985)

There Is No Polynomial Deterministic Space Simulation of Probabilistic Space with a Two-Way Random-Tape Generator

MAREK KARPINSKI*

Department of Computer Science, University of Pittsburgh, Pittsburgh, Pennsylvania 15213

AND

RUTGER VERBEEK

Department of Computer Science, University of Bonn, Bonn, West Germany

We prove there is no polynomial deterministic space simulation for two-way random-tape probabilistic space (Pr_2SPACE) (as defined in Borodin, A., Cook, S., and Pippenger, N. (1983) *Inform. Control* **58** 113–136) for all functions $f: \mathbb{N} \to \mathbb{N}$ and all $\alpha \in \mathbb{N}$, $Pr_2SPACE(f(n)) \nsubseteq DSPACE(f(n)^\alpha)$. This is the answer to the problem formulated in op cit., whether the deterministic squared-space simulation (for recognizers and transducers) generalizes to the two-way random-tape machine model. We prove, in fact, a stronger result saying that even space-bounded Las Vegas two-way random-tape algorithms (yielding always the correct answer and terminating with probability 1) are exponentially more efficient than the deterministic ones.

INTRODUCTION

Jung (1981) and Borodin, Cook, and Pippenger (1983) prove that both the probabilistic acceptors and transducers working in space $f(n) \geqslant \log n$ can be simulated in deterministic $f(n)^2$ space. The definition of probabilistic Turing machines uses a one-way read-only random tape. The model of probabilistic machine (Gill, 1977) may be viewed as a deterministic machine with a one-way only access to the random bits sequence. A two-way random tape proposed in Borodin, *et al.* (1983) allows multiple access to the random bits sequence which is stored on the two-way read-only tape. The problem posed in Borodin, *et al.* (1983) whether the $f(n)^2$-

* Supported by the Department of Computer Science, Carnegie–Mellon University, Pittsburgh, PA 15213.

0019-9958/85 $3.00

deterministic space simulation holds also for the two-way random-tape ($Pr_2SPACE(f(n))$).

Let $\psi \subseteq \Sigma^* \times \{0, 1\}^\omega$ be a binary predicate, where $\psi(x, y)$ is computed by a deterministic machine M with two two-way read-only input tapes. If M stops on an initial segment of y, then $\psi(x, y)$ is defined. $x \in \Sigma^*$ is recognized by M if and only if $\Pr\{\psi(x, y) = \text{true}\} > \frac{1}{2}$. We call M a probabilistic machine (over the alphabet Σ) with two-way random tape. Let $L_M \subseteq \Sigma^*$ denote the set recognized by M. If M is $S(|x|)$ space bounded, then L_M belongs to the two-way random-tape probabilistic space $S(n)$, $L_M \in Pr_2SPACE(S(n))$. If in addition M is $T(|x|)$ time bounded, then $L_M \in Pr_2TISP(T(n), S(n))$. We say that L_M belongs to the two-way Las Vegas (Babai, Grigoryev, Yu, and Mount, 1982) space $S(n)$, $L_M \in \Delta_2SPACE(S(n))$, if for all $x \in \Sigma^*$ either $\Pr\{\psi_M(x, y) = \text{true}\} = 1$ or $\Pr\{\psi_M(x, y) = \text{false}\} = 1$.

We prove that the class of $\log f(n)$ space bounded Las Vegas algorithms with two-way random tape (terminating with probability 1 and yielding always the correct result) denoted by $\Delta_2SPACE(\log f(n))$ (time bounded Las Vegas algorithms are defined in Adleman and Manders (1982); Babai *et al.* (1982) are as powerful as $DSPACE(f(n))$. Therefore there is no polynomial simulation for this class, which answers the problem of Borodin *et al.* (1983).

REMARKS

1. This result is related to the recent result of Savitch and Dymond (1984) that "consistent" NSPACE is exponentially more powerful than DSPACE. The similarity becomes clear, if the reset mechanism in the original definition of consistent NSPACE is replaced by a two-way tape, on which the initial nondeterministic choices are stored. The proof of our Theorem 2 can be applied to this case.

2. The model of a probabilistic machine with two-way random tape may be viewed as a deterministic machine with a random oracle stored on a two-way tape. The oracle tape records the outcome of an infinite sequence of independent unbiased coin tosses. The classical model of Gill (1977) may be viewed as a deterministic machine with a random oracle stored on a one-way tape. The classical oracle machine (Bennett *et al.* (1981)) is a deterministic machine with oracle stored on a device resembling random-access store rather than tape (i.e., the question must be written on a query tape within the space bound). Denote by $DSPACE^{(A)_2}(f(n))$ the class of sets recognized by $f(n)$ space bounded deterministic Turing machines with oracle A stored on a two-way

tape. Then, with probability 1 (i.e., for almost all oracles), $\mathrm{DSPACE}^{(A)_2}(f(n)) \not\supseteq \Delta_2\mathrm{SPACE}(f(n))$ (the inequivalence results from the fact that, with probability 1, $A \notin \Delta_2\mathrm{SPACE}(f(n))$).

RESULTS

THEOREM 1. *For every function* $f\colon \mathbb{N} \to \mathbb{N}$,

$$\bigcup_{k \in \mathbb{N}} \Delta_2\mathrm{TISP}(2^{2^{2^{k \cdot \log f(n)}}}, \log f(n)) \supseteq \mathrm{DSPACE}(f(n)).$$

COROLLARY. *For every function* f,

$$\mathrm{Pr}_2\,\mathrm{SPACE}(\log f(n)) \supseteq \Delta_2\mathrm{SPACE}(\log f(n)) \supseteq \mathrm{DSPACE}(f(n)).$$

COROLLARY (Problem of Borodin *et al.* (1983)).

$$\mathrm{Pr}_2\,\mathrm{SPACE}(f(n)) \not\subseteq \mathrm{DSPACE}(f(n)^2).$$

Proof of Theorem 1. Suppose $\mathscr{T}$ is a $f(n)$ space bounded deterministic Turing machine with one work tape. Suppose that $\mathscr{T}$ stops on every input (see Sipser (1980)).

For $x \in \Sigma^*$, $\mathrm{comp}_{\mathscr{T}}(x) \in \bar{\Sigma}^*$ will denote the computation of $\mathscr{T}$ over x (not recording the input or input position). The probability that the random tape will contain as a subsequence ¢ $\mathrm{comp}_{\mathscr{T}}(x)$ \$, $x \in \Sigma^*$ (encoded as a binary sequence), is equal to 1. On the other hand, the set $\{(x, u\,¢\,\mathrm{comp}_{\mathscr{T}}(x)\,\$\,v) \mid x \in \Sigma^*, u, v \in \bar{\Sigma}^*\}$ is recognized by a $\log f(n)$ bounded deterministic Turing machine $\mathscr{M}$ with two input tapes (only the position in the current storage-configuration of $\mathscr{T}$ must be stored).

Take now this machine $\mathscr{M}$, put it on the random tape and let it search for ¢ $\mathrm{comp}_{\mathscr{T}}(x)$ \$. This string will appear on the random tape with probability 1. Thus $\mathscr{M}$ stops with probability 1 and gives the correct result (according to the halting configuration in $\mathrm{comp}_{\mathscr{T}}(x)$). The expected time for the simulation lies in

$$\bigcup_k (2^{k \cdot |\mathrm{comp}_{\mathscr{T}}(x)|}) \leqslant \bigcup_k (2^{f(|x|) \cdot 2^{k \cdot f(|x|)}}) \leqslant \bigcup_k (2^{2^{2^{k \cdot \log f(|x|)}}}).$$

Theorem 1 is valid also for transducers; in this case $\mathscr{M}$ begins outputting after it has found and verified $\mathrm{comp}_{\mathscr{T}}(x)$. ∎

THEOREM 2. *For every function* f,

$$\Delta_2\mathrm{SPACE}(f(n)) \subseteq \bigcup_k \mathrm{SPACE}(n^4 \cdot 2^{k \cdot f(n)}).$$

COROLLARY. *If $f(n) \geqslant \log n$, then*

$$\Delta_2\text{SPACE}(f(n)) = \bigcup_k \text{DSPACE}(2^{k \cdot f(n)}).$$

In particular,

$$\Delta_2\text{SPACE}(\log n) = \text{PSPACE}.$$

Proof of Theorem 2. Let $\mathscr{M}$ be an $f(n)$ bounded Δ_2 machine. A configuration of $\mathscr{M}$ contains the position on the input and the content of the work tape (but not the position on the random tape). The number of configurations accessible on input x is bounded by $|x| \cdot 2^{k \cdot f|x|}$.

$\mathscr{M}$ is simulated by a Δ_1-machine $\mathscr{T}$ (i.e., with one-way random tape) in the same way as a two-way finite automaton is simulated by a one-way FA (see Hopcroft (1979)). It holds a table which says for each pair of configurations: if $\mathscr{M}$ is in configuration c and goes left (on the random tape) then it can (or cannot) come back in configuration c'. In addition it is stored whether or not $\mathscr{M}$ starting in configuration c can go left and never come back (in this case it is stored whether $\mathscr{M}$ accepts or rejects).

It is easy to see that $\mathscr{T}$ uses $(|x| \cdot 2^{k \cdot f|x|})^2$ space for two such tables and that these tables are sufficient to determine whether $\mathscr{M}$ stops, and if it stops, to determine the decision. Since $\mathscr{M}$ never gives a wrong result, $\mathscr{T}$ accepts the same set as $\mathscr{M}$. Since $\Delta_1\text{SPACE}(f(n)) \subseteq \text{Pr SPACE}(f(n)) \subseteq \text{DSPACE}(f(n)^2)$ (Borodin *et al.* 1983) $\mathscr{T}$ can be simulated by a deterministic machine in $O(|x|^4 \cdot 2^{4k \cdot f|x|})$ space. ∎

We were not able to extend the upper bound of Theorem 2 to the case of probabilistic machines with non-zero error probability. It is even not known whether or not $\text{Pr}_2\,\text{SPACE}$ is Blum complexity measure (Blum, 1967).

OPEN PROBLEM

Is there a recursive function h, such that for every f

$$\text{Pr}_2\,\text{SPACE}(f(n)) \subseteq \text{DSPACE}(hf(n))?$$

Is every set recognized by a probabilistic *finite* automaton with two-way random-tape recursive, i.e., $\text{Pr}_2\,\text{SPACE}(O(1)) \subseteq \text{DSPACE}(h(n))$ for some recursive h?

(By Karpinski and Verbeek (1984) the set of *computations* can be recognized by probabilistic finite two-way automata with one-way random-type and bounded error probability).[1]

RECEIVED September 26, 1984; ACCEPTED February 21, 1985

REFERENCES

ADLEMAN, L., AND MANDERS, K. (1977), Reducibility, randomness and intractibility, *in* "Proc. 9th ACM Sympos. Theory of Comput.", pp. 151–163.

BLUM, M. (1967), A machine-independent theory of the complexity of recursive functions, *J. Assoc. Comput. Mach.* **4**, 322–336.

BORODIN, A., COOK, S., AND PIPPENGER, N. (1983), Parallel computation for well-endowed rings and space-bounded probabilistic machines, *Inform. Control* **58**, 113–136.

BENNETT, C., AND GILL, J. (1981), Relative to a random oracle A, $P^A \neq NP^A \neq co-NP^A$ with probability 1, *SIAM J. Comput.* **10**, 96–114.

BABAI, L., GRIGORYEV, D. YU., AND MOUNT, D. M. (1982), Isomorphism of Graphs with bounded eigenvalue multiplicity, *in* "Proc. 14th ACM Sympos. Theory of Comput.", 310–324.

GILL, J. (1977), Computational complexity of probabilistic Turing machines, *SIAM J. Comput.* **6**, 675–694.

HOPCROFT, J., AND ULLMAN, J. (1979), "Introduction to Automata Theory, Languages, and Computation," Addison–Wesley, Reading, Mass.

JUNG, H. (1981), Relationships betwen probabilistic and determinstic tape complexity, 10th MFCS, Lecture Notes in Comput. Sci. Vol. 118, pp. 339–346, Springer-Verlag, New York/Berlin.

KARPINSKI, M., AND VERBEEK, R. (1984), On the Monte Carlo space-constructible functions and separation results for probabilistic complexity classes, Interner Bericht I/3 des Inst. Informatik, Univ. Bonn.

SAVITCH, W., AND DYMOND, P. (1984), Consistency in nondeterministic storage, *J. Comput. System Sci.* **29**, 118–132.

SIPSER, M. (1980), Halting space bounded computations, *Theoret. Comput. Sci.* **10**, 335–338.

[1] *Note in proof.* Meanwhile the authors were able to solve this problem. The first function h mentioned above is in fact recursive and $2^{O(n)}$ and the second is $n^2 \log^2 n$. Therefore $\mathrm{Pr}_2\mathrm{SPACE}$ is a Blum complexity measure.